WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Friday, October 21, 2005

Hide?	Set Name	Query	Hit Count
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ		
☐	L15	L14 and (fixed criteria)	1
☐	L14	L9 and (record$ near2 action)	1249
☐	L13	L9 and record$	80706
☐	L12	L9 and (field same action class)	30
	DB=USPT; PLUR=YES; OP=ADJ		
☐	L11	'5157779'.pn.	1
☐	L10	'5157779'.pn.	1
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ		
☐	L9	multimedia or playback	139794
☐	L8	L1 and (field same class list)	9
☐	L7	L5 and (field same class list)	0
☐	L6	L5 and (field same action class)	0
☐	L5	(L4 or l3 or l2) and (multimedia or playback).ab.	4
☐	L4	717/113.ccls. and (multimedia or playback)	24
☐	L3	717/109.ccls. and (multimedia or playback)	45
☐	L2	717/105.ccls. and (multimedia or playback)	26
☐	L1	717/104-113,125,127.ccls.	2250

END OF SEARCH HISTORY

Hit List

First Hit | Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 9 of 9 returned.

☐ 1. Document ID: US 20050177816 A1

L8: Entry 1 of 9 File: PGPB Aug 11, 2005

PGPUB-DOCUMENT-NUMBER: 20050177816
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20050177816 A1

TITLE: Automatic generation of graphical program code for a graphical program based on the target platform of the graphical program

PUBLICATION-DATE: August 11, 2005

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY
Kudukoli, Ramprasad	Austin	TX	US
Gabbert, Adam K.	Austin	TX	US
Andrade, Hugo A.	Austin	TX	US
Novacek, Matthew E.	Austin	TX	US
Darowski, Lukasz T.	Urbandale	IA	US

US-CL-CURRENT: 717/105; 703/1, 716/11, 717/109, 717/136

ABSTRACT:

A system and method for programmatically generating a graphical program in response to receiving input, e.g., user or process input. The input may specify functionality of the graphical program to be generated, and also indicate a target platform. In response to the input, a graphical program implementing the specified functionality may be programmatically generated for execution on the indicated target platform. Thus, different graphical programs, or different implementations of the graphical program, may be generated, depending on the input received. The graphical program (or implementation) may be at least partly optimized for execution on the indicated target platform. The graphical program may include a block diagram portion and a user interface portion, where the block diagram portion is specified for execution on the target platform, and the user interface portion is specified for execution on a computer system coupled to the target platform, e.g., for display of a user interface.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw D

☐ 2. Document ID: US 20030167455 A1

L8: Entry 2 of 9 File: PGPB Sep 4, 2003

PGPUB-DOCUMENT-NUMBER: 20030167455
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20030167455 A1

TITLE: Automatic software production system

PUBLICATION-DATE: September 4, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY
Iborra, Jose	Denia Alicante		ES
Pastor, Oscar	Valencia		ES

US-CL-CURRENT: 717/105

ABSTRACT:

An automated software production system is provided, in which system requirements are captured, converted into a formal specification, and validated for correctness and completeness. In addition, a translator is provided to automatically generate a complete, robust software application based on the validated formal specification, including user-interface code and error handling code.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw D

☐ 3. Document ID: US 20020100014 A1

L8: Entry 3 of 9 File: PGPB Jul 25, 2002

PGPUB-DOCUMENT-NUMBER: 20020100014
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20020100014 A1

TITLE: Automatic software production system

PUBLICATION-DATE: July 25, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY
Iborra, Jose	Denia Alicante		ES
Pastor, Oscar	Valencia		ES

US-CL-CURRENT: 717/104; 717/105, 717/106, 717/136

ABSTRACT:

An automated software production system is provided, in which system requirements are captured, converted into a formal specification, and validated for correctness and completeness. In addition, a translator is provided to automatically generate a

complete, robust software application based on the validated formal specification, including user-interface code and error handling code.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw D

☐ 4. Document ID: US 20020062475 A1

L8: Entry 4 of 9 File: PGPB May 23, 2002

PGPUB-DOCUMENT-NUMBER: 20020062475
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20020062475 A1

TITLE: Automatic software production system

PUBLICATION-DATE: May 23, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY
Iborra, Jose	Denia Alicante		ES
Pastor, Oscar	Valencia		ES

US-CL-CURRENT: 717/108

ABSTRACT:

An automated software production system is provided, in which system requirements are captured, converted into a formal specification, and validated for correctness and completeness. In addition, a translator is provided to automatically generate a complete, robust software application based on the validated formal specification, including user-interface code and error handling code.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw D

☐ 5. Document ID: US 6748580 B1

L8: Entry 5 of 9 File: USPT Jun 8, 2004

US-PAT-NO: 6748580
DOCUMENT-IDENTIFIER: US 6748580 B1

TITLE: Method and apparatus for creating software tools using a JTML interface

DATE-ISSUED: June 8, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Sur; Robert K.	Austin	TX		
Price; Jeffrey K.	Austin	TX		

US-CL-CURRENT: 717/105; 715/513, 717/109, 717/113, 719/328

ABSTRACT:

A user creates a software tool using a simple Java text markup language (JTML) interface. The user provides the JTML server with JTML pseudo code (code objects) which the JTML server recognizes, and performs routine programming tasks associated with the JTML code objects. The JTML server builds the executable JTML tool by invoking Java classes, associated with the code object, which are necessary to create the tool. The Java classes also create the user interface necessary for the user to enter data and commands for executing the JTML tool once the tool has been created. The JTML server acts as a secure port to the host by insulating the host contact or connection from a client. No code or command passes directly between the host and the client, instead the JTML server mediates all transactions between the two. Additional security is provided by the JTML server logging each user's access and each user's transaction. A record is therefore created from all user interaction between the client and the JTML server for review.

27 Claims, 13 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 9

☐ 6. Document ID: US 6654948 B1

L8: Entry 6 of 9 File: USPT Nov 25, 2003

US-PAT-NO: 6654948
DOCUMENT-IDENTIFIER: US 6654948 B1

TITLE: Methods and apparatus for partial and consistent monitoring of object-oriented programs and systems

DATE-ISSUED: November 25, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Konuru; Ravi B.	Tarrytown	NY		
Gruber; Olivier	Jackson Heights	NY		
De Pauw; Wim	Scarborough	NY		

US-CL-CURRENT: 717/127; 719/318

ABSTRACT:

A technique for monitoring events generated by an object-oriented system comprises the steps/operations of: (i) monitoring events which describe executed operations associated with the object-oriented system; and (ii) applying one or more sequencing rules when reporting a subset of the monitored events, the one or more sequencing rules substantially ensuring consistent reporting of the subset of monitored events. Preferably, monitoring continues when event reporting is at least partially disabled. Further, the monitoring step/operation may include dividing the monitored events into categories. One category may include entity events, an entity

event defining an existence status of a given event. Another category may include activity events, an activity event defining an operation associated with a given event. Still further, the entity events and activity events may be further divided into at least one of an object event category, an execution event category, a type event category and a synchronization event category. The sequencing rules are applied to maintain substantial consistency with respect to information associated with the categories.

26 Claims, 6 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 4

☐ 7. Document ID: US 6226788 B1

L8: Entry 7 of 9 File: USPT May 1, 2001

US-PAT-NO: 6226788
DOCUMENT-IDENTIFIER: US 6226788 B1

TITLE: Extensible network management system

DATE-ISSUED: May 1, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Schoening; Charles B.	Guttenberg	NJ		
Smith, Jr.; Richard J.	Danville	CA		
Schleimer; Stephen I.	San Jose	CA		

US-CL-CURRENT: 717/107; 709/203, 717/108

ABSTRACT:

In a network management system, a method and apparatus for preparing a computer program for execution in relation to a particular network device among a plurality of network devices having a plurality of device types is provided. Each network device is associated with a device type value, and each network device has an associated device mapper. The device mappers are stored in a hierarchical structure that reflects a functional relationship or family relationship of the devices. Functions to be carried out by one or more devices are expressed as a plurality of executable program components. Preferably, each executable program component has one or more classes that define executable functions. Each device mapper associates a device type value with one or more overridden classes in the executable program components and one or more overriding classes. At runtime, device type values are acquired for each device in the managed network. For each device type, one or more functions are assembled using only the executable program components associated with that device type. Based on the device mapper of that device type, classes in the executable program components are overridden and the overriding classes are substituted. As a result, at runtime the network management system integrates into itself executable program components for new devices.

44 Claims, 36 Drawing figures

Exemplary Claim Number: 1
Number of Drawing Sheets: 35

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw D

☐ 8. Document ID: US 6226785 B1

L8: Entry 8 of 9 File: USPT May 1, 2001

US-PAT-NO: 6226785
DOCUMENT-IDENTIFIER: US 6226785 B1

TITLE: Method and apparatus for storing and replaying creation history of multimedia software or other software content

DATE-ISSUED: May 1, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Peterson; Alan R.	Palo Alto	CA		
Spohrer; James C.	Santa Clara	CA		

US-CL-CURRENT: 717/106; 715/500, 715/967

ABSTRACT:

An authoring or other software tool is provided which allows users of the tool to review the creation and evolution of all or part of a content such as a multimedia title created using a tool incorporating the invention. According to the invention, an authoring tool comprises a recording mechanism for recording actions performed during the development or modification of software content in a manner that preserves, directly or indirectly, the relative chronological order of the actions and a playback mechanism for performing or otherwise simulating recorded actions. Preferably, the playback of an action includes annotation using text, recorded or computer-generated voice, video and/or graphic animation. Preferably, an authoring tool embodying the invention allows an author to specify criteria for which actions are to be played back, thereby providing an author with a mechanism for focusing on a particular subset of the recorded actions.

41 Claims, 17 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 10

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw D

☐ 9. Document ID: US 5838973 A

L8: Entry 9 of 9 File: USPT Nov 17, 1998

US-PAT-NO: 5838973
DOCUMENT-IDENTIFIER: US 5838973 A

** See image for Certificate of Correction **

TITLE: System and method for interactively transforming a system or process into a visual representation

DATE-ISSUED: November 17, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Carpenter-Smith; Theodore R.	St. Charles	IL		
Gombar; Michael John	Elburn	IL		
Fisher; James B.	Naperville	IL		
Barfield; Thomas M.	El Grove	IL		

US-CL-CURRENT: 717/105; 345/418, 345/419, 709/205, 715/771, 715/961, 715/967, 717/108, 719/328

ABSTRACT:

A computerized modeling system is provided. The present invention is a computer-implemented, interactive, real-time software tool, for physically transforming a system or process into a visual representation. The software tool includes a class developer for interactively developing the visual representation in real time. The visual representation is physically embodied in a computer-readable medium for visualization on a computer display device, and includes at least one of a class, a class behavior, a class attribute, a collaboration, and a collaboration message flow. The software tool also includes a three-dimensional visual representation module for displaying a three-dimensional depiction of the visual representation on the display device.

101 Claims, 77 Drawing figures
Exemplary Claim Number: 36
Number of Drawing Sheets: 42